7N-61-CR
84788
P-29

FINAL REPORT

on

NAG-1-1045

A SPACE-MARCHING ALGORITHM FOR THE PARABOLIZED NAVIER-STOKES EQUATIONS WITH CHEMISTRY - PHASE 3

Robert W. Walters - Principal Investigator
Department of Aerospace and Ocean Engineering
Virginia Polytechnic Institute and State University

April 1992

(NASA-CR-190235) A SPACE-MARCHING ALGORITHM FOR THE PARABOLIZED NAVIER-STOKES EQUATIONS WITH CHEMISTRY, PHASE 3 (Virginia Polytechnic Inst. and State Univ.) 29 p

N92-70573

Unclas

Z9/61 0084788

Installation 1.1
3/30/92

This version of gasp is setup to run identically in a number of different unix environments. (Our main development environments are Cray Y-MP, and SGI workstations however, compilation requires the use of different compilation flags All Makefiles make use of the "include" option to incorporate machine dependent parameters.

The file Makefile.def in the main gasp directories contains all machine specific parameters. i.e. optimization flags, and compiler names are contained in this file. Several default configurations may be found in the Makedef directory. in order to extract the default machine configuration file simply type
make "machine type" for example
make iris
will copy the appropriate configuration file for ieee workstations
make cray5
is suitable for cray operating systems with cf77 vs 5.x (NAS Crays)
make cray4
is suitable for cray operating systems with cf77 vs 4.x (Langley Crays)
make convex
is suitable for convex operating systems

Note, the compilation flags are suitable for default optimization if debugging is desired modify the FFLAGS & CFLAGS variables from Makefile.def, it is not necessary to modify the Makefile in the source directory

Note: The makefiles in the source directories should not require modification for machine dependent parameters, only make modifications to Makefile.def to port to different hardware/ software environments

COMPILATION:
in order to compile the gasp execution environment, first select the machine appropriate for you, and type from the main gasp directory
'make MACHINENAME'
edit the Makefile.def file and make appropriate modification to the compiliation flags.
then type
'make install'
(this will compile the gasp CFD flow solver, print the post-processor and dbasemgr, the utility used to build the chemistry and thermodynamics database, finally, the chemistry and thermodynamics database will be created.
all executables will be installed in the bin directory.
everything is now ready to run your own problems or any of the test cases located in the samples subdirectory

input.bouns 1.1
3/30/92

boundary condition files

if any of the boundaries, I0, Idim, J0, Jdim, K0, Kdim have type 0, then a pointwise boundary condition file must be specified.

This file must contain a boundary condition flag for each face on the boundaries with bc type 0

```
note there are (jdim-1)*(kdim-1) faces on the I0, and Idim boundaries
               (kdim-1)*(idim-1) faces on the J0, and Jdim boundaries
               (idim-1)*(jdim-1) faces on the K0, and Kdim boundaries
```

the logic for reading these flags is

```
if( boundary condition for I0 boundary = 0) then

   read( from bci file) ((i0_boun_type(j,k),j=1,jdim-1),k=1,kdim-1)

end if

if( boundary condition for Idim boundary = 0) then

   read( from bci file) ((idim_boun_type(j,k),j=1,jdim-1),k=1,kdim-1)

end if

if( boundary condition for J0 boundary = 0) then

   read( from bci file) ((j0_boun_type(k,i),k=1,kdim-1),i=1,idim-1)

end if

if( boundary condition for Jdim boundary = 0) then

   read( from bci file) ((jdim_boun_type(k,i),k=1,kdim-1),i=1,idim-1)

end if

if( boundary condition for K0 boundary = 0) then

   read( from bci file) ((k0_boun_type(i,j),i=1,idim-1),j=1,jdim-1)

end if

if( boundary condition for Kdim boundary = 0) then

   read( from bci file) ((kdim_boun_type(i,j),i=1,idim-1),j=1,jdim-1)

end if
```

NOTE: the reading of the j0 and jdim boundaries is different from that in gaspl.x

if any of the boundaries face has bc type 2, then the q for that face must be read in from the boundary condition q file

Note, either the global boundary condition flags may be set to type 2, or the pointwise boundary conditions may be set to type 2.

the format for the q file is one set of q's per line
the q's must be specified in primitive variables in dimensional quantities appropriate for the iunits flag specified in the main input deck

first are all nspec densities, then u, v, and w, and finally pressure.

NOTE: the temperature is not required as with gasp 1.x

the following provides a flow chart of the input of the boundary q's

```
for k = 1 to kdim-1
   for j = 1 to jdim-1

      if( ( boundary condition for I0 boundary = 2 )
         or
          ( boundary condition for I0 boundary = 0 )
            and pointwise boundary condition for point j,k = 2 ) ) then

          read(from bc q file) (qbci0(j,k,ne),ne = 1,neqn)

       end if

   end for j
end for k

for k = 1 to kdim-1
   for j = 1 to jdim-1

      if( ( boundary condition for Idim boundary = 2 )
         or
          ( boundary condition for Idim boundary = 0 )
            and pointwise boundary condition for point j,k = 2 ) ) then

          read(from bc q file) (qbcidim(j,k,ne),ne = 1,neqn)

       end if

   end for j
end for k

for i = 1 to idim-1
   for k = 1 to kdim-1

      if( ( boundary condition for J0 boundary = 2 )
         or
          ( boundary condition for J0 boundary = 0 )
            and pointwise boundary condition for point k,i = 2 ) ) then

          read(from bc q file) (qbcj0(k,i,ne),ne = 1,neqn)

       end if

   end for k
end for i

for i = 1 to idim-1
   for k = 1 to kdim-1

      if( ( boundary condition for Jdim boundary = 2 )
         or
          ( boundary condition for Jdim boundary = 0 )
            and pointwise boundary condition for point k,i = 2 ) ) then

          read(from bc q file) (qbcjdim(k,i,ne),ne = 1,neqn)
```

```
      end if

   end for k
end for i

for j = 1 to jdim-1
   for i = 1 to idim-1

     if( ( boundary condition for K0 boundary = 2 )
        or
         ( boundary condition for K0 boundary = 0 )
          and pointwise boundary condition for point i,j = 2 ) ) then

         read(from bc q file) (qbck0(i,j,ne),ne = 1,neqn)

      end if

   end for i
end for j

for j = 1 to jdim-1
   for i = 1 to idim-1

     if( ( boundary condition for Kdim boundary = 2 )
        or
         ( boundary condition for Kdim boundary = 0 )
          and pointwise boundary condition for point i,j = 2 ) ) then

         read(from bc q file) (qbckdim(i,j,ne),ne = 1,neqn)

      end if

   end for i
end for j

NOTE: the reading of the j0 and jdim boundaries is different from
      that in gaspl.x
```

input.dbasemgr 1.1
3/30/92

DBASEMGR

Introduction:
DBASEMGR is a management program for the GASP databases. This program takes ordinary text files (called init files) and converts them into direct access files for use by GASP. This document describes how to install, run, and configure the database manager

Quick Installation & Execution:
When you first make gasp, the database manager will also be compiled and run for the first time. During this initial run it will create, summarize, and perform extended error analysis on the databases. You will see messages from the program during each phase of the execution. Seven files are created during the initial execution, the three databases, three summary files, and a log file. Each of these files are explained later in this documentation.

Configuration:
DBASEMGR has several options which tell the program what to do. The options are specified in the file dbasemgr.cnf. This file must reside in the current working directory (the directory where the init files reside). If this file does not exist then the program will create a default configuration file and use the default values during execution. It is recommended that you allow DBASEMGR to create a default configuration file and then edit the file to change the way the program operates.

The configuration file has the following format:

```
Line 1:  Title (ignored)
Line 2:  Title (ignored)
Line 3:  Names of the reaction, species, and models init (ascii) files.
Line 4:  Title (ignored)
Line 5:  Names of the reaction, species, and models database files.
Line 6:  Title (ignored)
Line 7:  Names of the reaction, species, and models summary files.
Line 8:  Title (ignored)
Line 9:  Names of the reaction, species, and models addition files.
Line 10: Title (ignored)
Line 11: Names of the reaction, species, and models extraction files.
Line 12: Title (ignored)
Line 13: Flag for creating the Liu files, the names of the Liu ascii
         file and binary file respectively.
Line 14: Title (ignored)
Line 15: Name of the log file for errors.
Line 16: Title (ignored)
Line 17: Overwrite variable.
Line 17: Title (ignored)
Line 18: Mode which tells the program what to do next.
         There can be up to 10 modes, each on a separate line.
```

Files:
[ASCII Init Files]
The "init" files are the initialization files used by DBASEMGR to create the binary databases for GASP. The init files are broken down into three separate files, one for the reactions, one for the species, and one for the models.

The format of the REACTIONS INIT file is as follows:

```
  Line 1:  Title (record separator - ignored)
  Line 2:  Title (ignored)
  Line 3:  Reaction number and reaction in symbolic form
  Line 4:  Title (ignored)
  Line 5:  Rate calculation flag and reference
  Line 6:  Title (ignored)
  Line 7:  Forward rate coefficients
  Line 8:  Title (ignored)
  Line 9:  Equilibrium rate coefficients
  Line 10: Title (ignored)
  Line 11: Title (ignored)
  Line 12: Title (ignored)
  Line 13: Title (ignored)
  Line 14: Title (ignored)
  Line 15: Stoichiometric Coefficients (Left Hand Side)
  Line 16: Stoichiometric Coefficients (Right Hand Side)
```

The rate calculation flag is equivalent to icheq in GASP. The following values are currently supported:

```
  icheq = 1  Arrhenius equilibrium rate
  icheq = 2  Park equilibrium rates
  icheq = 3  Equilibrium rate calculated from LeRC curve fits from
             Gibbs free energy.
```

All forward rates are calculated using Arrhenius form.
For icheq = 3, line 9 is treated as a dummy title.

The format of the SPECIES INIT file is as follows:

```
Line 1:  Title (record separator - ignored)
Line 2:  Title (ignored)
Line 3:  Species number
Line 4:  Title (ignored)
Line 5:  Species symbol and molecular weight
Line 6:  Title (ignored)
Line 7:  Control flags (ibmu, ismu, isk, ilrc, ivib)
Line 8:  Title (ignored)
Line 9:  Title (ignored)
Line 10:
    .   :
    .   : LeRC Coefficients (Lower Range)
    .   :
Line 19:
Line 20: Title (ignored)
Line 21:
    .   :
    .   : LeRC Coefficients (Upper Range)
    .   :
Line 30:
Line 31: Title (ignored)
Line 32: Blottner coefficients for laminar viscosity
Line 33: Title (ignored)
Line 34: Sutherland coefficients for laminar viscosity
Line 35: Title (ignored)
Line 36: Sutherland coefficients for thermal conductivity
Line 37: Title (ignored)
Line 38: Number of vibrational temperatures
Line 39: Title (ignored)
Line 40: Vibrational Temperatures (All on single line)
```

The control flags specified on line 7 allow one or more of the coefficients to be left out for a given species. For example, if ibmu is set to zero then lines 31-32 should not be present. The control flags have the following meaning:

```
    ibmu - read Blottner viscosity coefficients
    ismu - read Sutherland viscosity coefficients
    isk  - read Sutherland thermal conductivity coefficients
```

ilrc - read Lewis Research coefficients (McBride coefficients)
ivib - read vibrational temperatures

The format of the MODELS INIT file is as follows:

Line 1: Title (record separator - ignored)
Line 2: Title (ignored)
Line 3: Model number and model name
Line 4: Title (ignored)
Line 5: Number of species and reactions for the model
Line 6: Title
Line 7: List of species in model. Each species is specified by the number in the species init file. Each must be placed on a separate line.
Line 8: Title (ignored)
Line 9: List of reactions in model. Each reaction is specified by the number in the reactions init file. Each reaction must be placed on a separate line.

[Database Files]
These are the files that are read in and used by GASP to retrieve information about a given model, species, or reaction.

[Summary Files]
The summary files contain useful information about the different databases. There are three summary files, one for each database.

[Extraction Files]
The extraction files are the same as the init files except that these are created from the binary database files. These files can be used as the init files after a record has been added to one of the databases (See adding species for more details)

[Report Log]
This is where all errors are logged. If the program complains about a possible error during the analysis phase, it will tell you to look in this file for more details. At the end of the file is a list of the models with the valid thermodynamic, laminar viscosity, and laminar thermal conductivity models for each model.

Modes:
There are 10 different modes that are valid.

```
help ...................... Prints listing of all valid modes.
                            If an invalid mode is chosen this
                            message will be printed.
create .................... Create GASP databases
summary ................... Print summaries of the databases
summarize ................. Same as summary
extract ................... Extracts the init files from the databases
analyze ................... Performs extended error analysis
analysis .................. Same as analyze
add-species ............... Add species to databases
add-rxn ................... Adds reactions to the databases
add-model ................. Adds models to the databases
```

Warning!
Adding species:
Adding species should always be handled by DBASEMGR. Since the other two databases are dependent on the species database these databases will have to be modified to reflect the addition of a species. When a species is added with DBASEMGR, the reactions and models database are modified to reflect the changes in the species database. After adding species it is recommended that new init files be created using the extract mode. The new init files can then be edited to add models and reactions.

```
  input.main 1.1
  3/30/92

&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&
&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&

Terminology used in GASP Input Decks:

&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&
&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&

Zone - a section of the computational domain that is described by an
       individual zonal input deck

Zonal Boundary - a physical boundary between two zones across which
                 information is passed

Group - a group of zones that are run simultaneously because
        solution information is shared between the zones in that
        group during the iteration process of the group.

Sequence level - the level of coarsening for a group of zones

Block - A block is described as a given group on a given sequence level

Marching - space marching which performs an iteration on a single plane
           of a given group at a time

Global iteration - performs an iteration on all planes of a given
                   group at a time


GASP has multiple input decks.  It has a main controller input deck and
also individual input decks for each zone in a given problem. Below is
a brief description of the input decks.

&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&
&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&

                  Main input deck description

&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&
&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&

The main input deck is broken down into 5 main sections:

    1) Main info - describes the problem as a whole
                      a) reference quantities
                      b) file control parameters
                      c) number of zones, zonal boundaries, groups

    2) Zone info - describes zone dependent information
                      a) zone file names
                      b) surrounding zonal boundaries

    3) Zonal Boundary info - describes zonal boundary information
                      a) grid alignment for boundaries
                      b) variable mapping for boundaries

    4) Group info - describes group dependent information
                      a) starting & ending zones
                      b) marching vs global iteration
                      c) mesh sequencing levels

    5) Block info - describes solution process
```

a) group & sequence level
b) # of cycles & convergence criteria
c) sweeping information

Each of these sections is divided in this file by a line that has the section name surrounded by #'s.

```
############################  MAIN INFO  ###############################

Each of the lines in this section do NOT repeat

************************* main info line 1 ***************************

     title card

----------------------------------------------------------------------
---------------------------- DUMMY LINE ------------------------------
----------------------------------------------------------------------

************************* main info line 2 ***************************

     iunits   -  unit specification flag
              = 0  English Units
              = 1  SI Units

     rhoref   -  reference density  (slugs/ft^3 or kg/m^3)

     vref     -  reference velocity in (ft/s  or  m/s)

     tref     -  reference temperature in (degrees R or Degrees K)

     lref     -  reference length (ft or m)

     NOTE: These are only used for non-dimensionalization in GASP.
           They are not related to the freestream values. You may
           choose the reference quantities any way you want. If
           you set them all to 1., then the calculations are all
           performed with dimensional variables. You may choose to
           set them such that your non-dimensional freestream
           variables are order one.

     NOTE: The grids  you provide must be non-dimensional thus if
           your grid is in feet and iunits = 0 select lref = 1.
           however if you grid is in yards lref must be set
           to the foot equivalent of 1 yard or lref = 3.


----------------------------------------------------------------------
---------------------------- DUMMY LINE ------------------------------
----------------------------------------------------------------------

************************* main info line 3 ***************************

     irest - flag for restart
           = 0 no restart
           = 1 restart

     memmode - memory mode
           = 0 low memory mode, use minimum memory at expense of i/o
           = 1 medium memory mode, middle of the road trade-off
                                   between low & high
           = 2 high memory mode, use enough memory to minimize i/o

     filmode - format of grid and restart files
           = 0  c - binary
```

```
        = 1  fortran formatted binary, direct access

    resmode - level of restart files
          positive numbers indicate all residual information to
          a single file
          negative numbers indicate residual information to individual
          files

          = 0 no residual information
          = +- 1    - output group residual information
          = +- 2    - output group and zone information
          = +- 3    - output group zone and sweep information
          = +- 4    - output group, zone, sweep, and plane info

-----------------------------------------------------------------------
---------------------------- DUMMY LINE -------------------------------
-----------------------------------------------------------------------

************************* main info line 4 ***************************

    thermo-chemistry database path name
      this is the pathname of the database describing thermodynamics
      and chemistry models

    This may be a relative or absolute path

    Waring: Pathname must be enclosed in 'single quotes'

-----------------------------------------------------------------------
---------------------------- DUMMY LINE -------------------------------
-----------------------------------------------------------------------

************************* main info line 5 ***************************

    nzone - total number of zones in domain

    nzboun - total number of zonal boundaries in domain

    ngroup - number of groups in the domain
            each group contains all zones which must be
            iterated upon simultaneously

    nblock - number of blocks to solve on
             each block represents a group & sequence level

    iblst - block to begin a run of the code on

    iblend - block to end a run of the code on

************************* end of main info ***************************

###########################  ZONE INFO  ##############################

-----------------------------------------------------------------------
---------------------------- DUMMY LINE -------------------------------
-----------------------------------------------------------------------

Each of the following lines in this section is repeated NZONE times

-----------------------------------------------------------------------
---------------------------- DUMMY LINE -------------------------------
-----------------------------------------------------------------------

************************* zone info line 1 ***************************

    zone # - zone #'s in increasing order
```

```
     zone file - file name for zonal input deck
     enclose file name in 'single quotes'

     filedir - storage direction for zonal grid & restart files
            = 1   -  grid & restart stored (jdim,kdim,idim)
            = 2   -  grid & restart stored (kdim,idim,jdim)
            = 3   -  grid & restart stored (idim,jdim,kdim)

     gridfil - file name for zonal grid file
     enclose file name in 'single quotes'

     restfil - file name for zonal restart file
     enclose file name in 'single quotes'

---------------------------------------------------------------------
---------------------------- DUMMY LINE ----------------------------
---------------------------------------------------------------------

************************* zone info line 2 ***************************

     zone # - zone #'s in increasing order

     fillzb - flag for initializing a zone with a given zonal boundary
            = 0 do not initialize in this way
            = 1 initialzes with i0 zonal boundary
            = 2 initialzes with idim zonal boundary
            = 3 initialzes with j0 zonal boundary
            = 4 initialzes with jdim zonal boundary
            = 5 initialzes with k0 zonal boundary
            = 6 initialzes with kdim zonal boundary

     # surr. zones - # of zonal boundaries around a given zone

     zb1,zb2,...zbn - the corresponding numbers of the zonal boundaries
                      (in list above) that surround a given zone.

************************* end of zone info ***************************

#######################  ZONAL BOUNDARY INFO ###########################

---------------------------------------------------------------------
---------------------------- DUMMY LINE ----------------------------
---------------------------------------------------------------------

The next SET of lines repeats for each zonal boundary (NZBOUN times)

---------------------------------------------------------------------
---------------------------- DUMMY LINE ----------------------------
---------------------------------------------------------------------

---------------------------------------------------------------------
---------------------------- DUMMY LINE ----------------------------
---------------------------------------------------------------------

********************** zonal boundary line 1 *************************

     zone# - zone # on one side of zonal boundary

     type - type of boundary for that zone
          = 1 - i0 boundary
          = 2 - idim boundary
          = 3 - j0 boundary
          = 4 - jdim boundary
          = 5 - k0 boundary
          = 6 - kdim boundary
```

```
     dir1 - direction (i,j,k) of grid in zone that corresponds to
            dir1 of zone on the other side of zonal boundary
          = 1 - i direction
          = 2 - j direction
          = 3 - k direction

     start1 - starting value for dir1 variable

     end1 - ending value for dir1 variable

     dir2 - direction (i,j,k) of grid in zone that corresponds to
            dir2 of zone on the other side of zonal boundary
          = 1 - i direction
          = 2 - j direction
          = 3 - k direction

     start2 - starting value for dir2 variable

     end2 - ending value for dir2 variable

-----------------------------------------------------------------------
---------------------------- DUMMY LINE ------------------------------
-----------------------------------------------------------------------

************************ zonal boundary line 2 ************************

     # var - # of state varibles for zone

     mapping - mapping of state variables from other zone into this one
               (List should be # var long)

-----------------------------------------------------------------------
---------------------------- DUMMY LINE ------------------------------
-----------------------------------------------------------------------

************************ zonal boundary line 3 ************************

     Same as zonal boundary line 1 using zone on the
     other side of the zonal boundary.

-----------------------------------------------------------------------
---------------------------- DUMMY LINE ------------------------------
-----------------------------------------------------------------------

************************ zonal boundary line 4 ************************

     Same as zonal boundary line 2 using zone on the
     other side of the zonal boundary.

********************* end of zonal boundary info **********************

###########################  GROUP INFO  ###############################

-----------------------------------------------------------------------
---------------------------- DUMMY LINE ------------------------------
-----------------------------------------------------------------------

The next SET of lines is repeated for each group (NGROUP times)

-----------------------------------------------------------------------
---------------------------- DUMMY LINE ------------------------------
-----------------------------------------------------------------------

-----------------------------------------------------------------------
---------------------------- DUMMY LINE ------------------------------
```

```
-----------------------------------------------------------------

************************* group info line 1 **************************

     starting zone - first zone in this group

     ending zone - last zone in this group

       NOTE: zones must be ordered such that the zone numbers
             for each group are sequential.


     imarch - space marching flag
            = 0 solve using global iterations
            = 1 space march, Stop calculation if tolerances on a plane
                             are not met in nit iterations.
            =-1 space march, do NOT stop on an i plane if tolerances
                are not me.  Proceed to next i plane and continue
                calculation.

     nseq - number of levels of mesh sequencing performed for the group

       NOTE: Every group must have nseq >= 1, if no sequencing is
             being performed, set nseq = 1

     *******************************************************
     the following subset is repeated nseq times in each group
     *******************************************************

     -----------------------------------------------------------
     ----------------------- DUMMY LINE ------------------------
     -----------------------------------------------------------

     -----------------------------------------------------------
     ----------------------- DUMMY LINE ------------------------
     -----------------------------------------------------------

     ****************** sequence info line 1 *******************

          This sequence info line is repeated for each zone in
          the group (from starting zone to ending zone)

          zone # - zone #'s in increasing order

          ilevel - level of coarsening in i-direction

          jlevel - level of coarsening in j-direction

          klevel - level of coarsening in k-direction

          NOTE : The sequencing is performed separately in
                 each direction.  A coarsening level of 1
                 corresponds to the finest mesh (or NO mesh
                 sequencing). A level of 2 corresponds to
                 every other grid line. A level of 3 corresponds
                 to every third grid line, and so on.  In
                 order to coarsen a given direction to
                 a certain level, the following constraint
                 on the grid dimension must apply :
                 (dimension-1) must be exactly divisible
                 by the coarsening level. For example:
                 the remainder of (idim-1)/ilevel must be
                 zero.

          NOTE : Sequence Level #1 MUST correspond to the fine
                 mesh, or ilevel = 1, jlevel = 1, klevel = 1
```

for all zones in this group

The terminology for this section is a bit confusing so here's an example :

```
sequence # 1
 zone #    ilevel  jlevel  klevel
   1         1       1       1
sequence # 2
 zone #    ilevel  jlevel  klevel
   1         4       3       1
sequence # 3
 zone #    ilevel  jlevel  klevel
   1         4       3       2
```

In this example there are 3 sequencing levels (labeled #1, #2, & #3). Note that the higher the sequencing level, the coarser the mesh. The individual coarsening capability for each direction is apparent with the random choices shown. In going from sequence level 1 to 2 the mesh is coarsened 4, 3, and 0 times in the i, j, and k directions repectively. In going from sequence level 2 to 3 the mesh is coarsened 0, 0, and 2 times in the i, j, and k directions respectively since the coarsening level is relative and always refers back to the finest mesh.

******************* end of sequence info *******************

************************ end of group info *****************************

########################### BLOCK INFO ###############################

---------------------------- DUMMY LINE ----------------------------
----

The next SET of lines is repeated for each block (NBLOCK times)

---------------------------- DUMMY LINE ----------------------------
----

---------------------------- DUMMY LINE ----------------------------
----

************************* block info line 1 **************************

igroup - group number for block

iseq - sequence level for block

nsweep - the number of different iteration types for this block. (note there must be at least one sweep for each zone in this block)

ncycle - number of cycles through each sweep

nwres - number of cycles performed in medium or high memory mode before the restart files are updated

rtolrg - normalized residual convergence criteria for this block rtolag - absolute residual convergence criteria for this block

----
---------------------------- DUMMY LINE ----------------------------
----

Sweeping information is a subsection of the block info section. The following line is repeated NSWEEP times

****************** sweep info line 1 **********************

sweep - sweep # in increasing order

zone - zone number corresponding to this sweep

isweep - time integration sweep direction
= -/+ 1 - sweep in -/+ i-direction
= -/+ 2 - sweep in -/+ j-direction
= -/+ 3 - sweep in -/+ k-direction

NOTE: if space marching, isweep must be positive

nit - number of iterations to perform this sweep before continuing to the next sweep

istart - i/j/k dimension to start this sweep

iend - i/j/k dimension to end this sweep

****************** end of sweep info **********************

************************* end of block info **************************

input.print 1.1
3/30/92

```
&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&
&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&

                    Print input deck description

&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&
&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&

Print consolidates all post processing functions into a single
integrated application. The following describes the print input
deck read from standard input

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
               Global Print Information, valid for all print lines
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

*************************** global line one ***************************

     title card

*************************** global line two ***************************

     Name of main gasp input deck for the problem

          NOTE:   Must be enclosed in 'single quotes'

----------------------------------------------------------------------
---------------------------- DUMMY LINE ----------------------------
----------------------------------------------------------------------

************************** global line three **************************

         Reference quantities: used for nondimensionalization in print

           Note: all reference quantities must be non-dimensionalized
                 to reference lenhth specified in main gasp input deck

           lenref - reference length to be used in moment calculation
           arearef - reference area to be used in force coefficient

           x0,y0,z0, reference point for taking moments

----------------------------------------------------------------------
---------------------------- DUMMY LINE ----------------------------
----------------------------------------------------------------------

************************** global line four **************************

     nline   - number of lines/sets of output

     iprtinf - flag for printing condition at infinity in line output

             = 0 no print out
             = 1 print out free stream quantities at top
                 of standard output
             = 2 print out free stream quantities before each
                 line/set to standard ouput

      memmode -  print memory mode flat

              = 0 low memory (Use this when printing multi-zone
                              plot3d type files)
```

```
            = 1 high memory mode, use this for most efficient
                line output calculation

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
         The remaining set of lines is repeated nline times
         Describe the individual line/sets to be printed
         each group of input is repeated nline time
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

----------------------------------------------------------------------
---------------------------- DUMMY LINE ----------------------------
----------------------------------------------------------------------

----------------------------------------------------------------------
---------------------------- DUMMY LINE ----------------------------
----------------------------------------------------------------------

******************** line/set specification line 1 ********************

     iprint  - flag for determining type of ouput file
             = 1 line output
                 prints to standard output, this output is
                 suitable for display with 80 columns

             = 2 Tab delineated output
                 prints each individual line to separate file
                 each line prints variables in tab delineated
                 format with one column per variable.
                 Note, this is suitable for input into a
                 line plotting package or spreadsheet type
                 analysis package (i.e. Kaliedagraph on Mac)

             = 3 Integrated quantities use this option to
                 print Cx, Cy, Cz, Cmx, Cmy, etc. to standard output
                 (note, you must use this option in conjuction with
                  either inode = 2,3, or 4)

             = 4 TECPLOT output

             = 5 PLOT3D ascii
                 each line is treated as a separate grid in a
                 plot3d ascii multi-grid format
                 read/mg/form
                 all plot3d ascii solutions written to file q.p3da
                 a grid file is created by default labeled grid.p3da

             = 6 PLOT3D fortran binary
                 each line is treated as a separate grid in a
                 plot3d unformatted multi-grid format
                 read/mg/unf
                 all plot3d unformatted solutions written to q.p3du
                 a grid file is created by default labeled grid.p3du
                 (Note a conversion from gasp precision to machine
                  single precision is performed i.e. on workstation
                  real*8 are converted to real*4)

             = 7 PLOT3D c binary
                 each line is treated as a separate grid in a
                 plot3d binary multi-grid format
                 read/mg/bin
                 all plot3d binary solutions written to file q.p3db
                 a grid file is created by default labeled grid.p3db
                 (Note a conversion from gasp precision to machine
                  single precision is performed i.e. on workstation
                  real*8 are converted to real*4)
```

```
          = 8 PLOT3D iris c binary (only supported on cray)
              each line is treated as a separate grid in a
              plot3d binary multi-grid format
              for use on ieee workstations
              read/mg/bin
              all plot3d binary solutions written to file q.p3dbi
              a grid file is created by default labeled grid.p3dbi
              (this option converts cray real*8 to ieee real*4)

              NOTE: this is the most efficient way to use
                    plot3d or fast on a workstation from solution
                    created on a cray

    iout    - flag for dimensionalizing output
            = 0 dimensional output (English units)
            = 1 dimensional output (SI units)
            = 2 non-dimensionalized by free stream
            = 3 non-dimensionalized by reference quantities

            NOTE: iout = 0, and iout = 1 are independant from the
                  iunits flag in the main gasp input deck
                  thus if you did the original calculation in SI
                  you can still view your solution in English units

    inode   - flag for determining location of output
            = 0 output grid & q's at cell centers
            = 1 output grid & q's at grid nodes
            = 2 output quantities on i faces
            = 3 output quantities on j faces
            = 4 output quantities on k faces
c
            NOTE: inode = 2,3,4 currently MUST ONLY be used with
                  ivar = 601,602,603,611,612,613,621,622,623
                         701,702,703,711,712,713,721,722,723

    zone    - zone number of output for that line

    seq -     sequence number for output for that particular zone

--------------------------------------------------------------------
--------------------------- DUMMY LINE ----------------------------
--------------------------------------------------------------------

******************* line/set specification line 2 ******************

    this input line specifies the range of output for the specific
    line.
    valid ranges
      inode = 0 (cell centered)
         [0:idim][0:jdim][0:kdim]
         note 0 and dim are at the boundaries
      inode = 1 (nodal output)
         [1:idim][1:jdim][1:kdim]
         note 1 and dim are at the boundaries
      inode = 2 (i faces)
         [1:idim][1:jdim-1][1:kdim-1]
      inode = 3 (j faces)
         [1:idim-1][1:jdim][1:kdim-1]
      inode = 4 (k faces)
         [1:idim-1][1:jdim-1][1:kdim]

    i1      - starting i indices for output
    i2      - ending i indices for output
    iinc    - i increment for output

    j1      - starting j indices for output
```

```
   j2       - ending j indices for output
   jinc     - j increment for output

   k1       - starting k indices for output
   k2       - ending k indices for output
   kinc     - k increment for output

   Note: Regardless of the mesh sequencing level, dimensions
         given to this line are with respect finest mesh

---------------------------------------------------------------------
---------------------------- DUMMY LINE -----------------------------
---------------------------------------------------------------------

******************* line/set specification line 3 *******************


     number of variables to be printed

     list of variables to be printed (numbers correspond to plot3d
                                      function numbers where applicable)

     supported functions

     1    x coordinate
     2    y coordinate
     3    z coordinate

    50    laminar viscosity
    51    laminar thermal conductivity

    60    mixture gamma

    70    Turbulent Kinetic Energy (K) per unit mass
    71    TKE dissipation rate (Epsilon) per unit mass
    72    Turbulent Kinetic Energy (Rho K) per unit volume
    73    TKE dissipation rate (Rho Epsilon) per unit volume

   100    Mixture Density

   110    Pressure
   114    Pressure Coefficient

   120    Temperature

   132    Stagnation Enthalpy per unit mass

   142    Stagnation Energy per unit mass

   150    U Velocity
   151    V Velocity
   152    W Velocity
   154    Mach Number
   155    Speed of Sound

   160    X-Momentum
   161    Y-Momentum
   162    Z-Momentum
   163    Stagnation Energy per unit volume

   501    Skin Friction Coefficient on I faces
   502    Skin Friction Coefficient on J faces
   503    Skin Friction Coefficient on K faces

   Note 501,502,503 are defined as the coefficient due to
                    the magnitude viscous forces tangent
```

to the I,J, or K faces

```
511   Heat Transfer Coefficient on I faces
512   Heat Transfer Coefficient on J faces
513   Heat Transfer Coefficient on K faces


601   total force on a face in the x direction
602   total force on a face in the y direction
603   total force on a face in the z direction

611   Pressure force on a face in the x direction
612   Pressure force on a face in the y direction
613   Pressure force on a face in the z direction

621   Viscous force on a face in the x direction
622   Viscous force on a face in the y direction
623   Viscous force on a face in the z direction

      Note: 601,602,603,611,612,613 are only valid
            for inode = 2,3,or 4
```

*********************** end of print input ***************************

```
input.zonal 1.1
3/30/92

&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&
&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&

             Zonal input deck, (one for each zone)

&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&
&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&

The input deck for each zone is organized into 5 basic sections

     1) Problem Specification
            a) grid size, # species, etc
            b) free stream quantities
            c) boundary condition information

     2) Time Integration
            a) time integration technique
            b) cfl number
            c) reuse of LU decomp parameters
            d) marching support parameters

     3) Inviscid Fluxes
            a) type of flux & Jacobian used
            b) variable interpolation & limiters

     4) Viscous Fluxes
            a) flux & Jacobian control
            b) transport property model parameters
            c) Prandtl & Schmidt numbers
            d) turbulence parameters

     5) Chemistry & Thermodynamics
            a) chemistry & thermodynamic model
            b) frozen, equil., & finite rate source term & Jacobian
               control parameters

######################## PROBLEM SPECIFICATION ########################


******************** problem specification line 1 *********************

     title card

---------------------------------------------------------------------
---------------------------- DUMMY LINE -----------------------------
---------------------------------------------------------------------

******************** problem specification line 2 *********************


     idim   - grid size in i direction

     jdim   - grid size in j direction

     kdim   - grid size in k direction

     nspec  - number of chemical species

     nnev   - number of nonequilibrium energy contributions

     NOTE:  grid dimension correspond to finest mesh in mesh sequencing
            coarser grid dimensions are determined from mesh sequencing
```

```
        information

----------------------------------------------------------------------
---------------------------- DUMMY LINE ------------------------------
----------------------------------------------------------------------

******************** problem specification line 3 *********************

     icond  - input initialization flag
            = 1 velocity components and temperature are input
            = 2 Mach number components and temperature are input
            = 3 velocity components and pressure are input
            = 4 Mach number components and pressure are input

     U,V,W  - free stream velocity components
              (units of velocity must correspond to selection of
               iunits flag in the main input deck)

   Mx,My,Mz - free stream Mach number components

     T,p    - free stream temperature/pressure
              (units of pressure or temperature must correspond to
               selection of iunits flag in the main input deck)

     xkiv   -  free stream value of turbulent intensity
               Kinf = (xkiv*Vinf)^2
              (units are important)

     tkelref - reference length for calculation of free stream epsilon
               or turbulent reynolds number.
              > 0 reference length input
              < 0 turbulent reynolds number input (L=-tkelref/xkiv/Re)
               eps = K^(3/2)/L, also eddy viscosity = rho*sqrt(K)*L.
              (units are important)

----------------------------------------------------------------------
---------------------------- DUMMY LINE ------------------------------
----------------------------------------------------------------------

******************** problem specification line 4 *********************

     Freestream species densities (nspec of them) (with proper units)

----------------------------------------------------------------------
---------------------------- DUMMY LINE ------------------------------
----------------------------------------------------------------------

******************** problem specification line 5 *********************

     impbc - implicit boundary condition flag
           = 0  -  explicit boundary conditions
           = 1  -  implicit boundary conditions

     initbc - initialization flag for boundary conditions
             this is only used if the corresponding boundary
             condition has type 2 values where q's are read
             in from a file.  It allows the domain to be
             initialized with profiles that are read in this way.
           = 0  -  do nothing
           = 1  -  initialize q's zone with i0 boundary values
           = 2  -  initialize q's zone with idim boundary values
           = 3  -  initialize q's zone with j0 boundary values
           = 4  -  initialize q's zone with jdim boundary values
           = 5  -  initialize q's zone with k0 boundary values
           = 6  -  initialize q's zone with kdim boundary values
```

i0bc - i0 boundary condition flag

idimbc - idim boundary condition flag

j0bc - j0 boundary condition flag

jdimbc - jdim boundary condition flag

k0bc - k0 boundary condition flag

kdimbc - kdim boundary condition flag

boundary conditions :

```
       = 0   do a point by point description of boundary
       = 1   fixed at the freestream conditions
       = 2   fixed at the values read from a file
       = 3   first order extrapolation
       = 4   second order extrapolation
       = 5   characteristic b.c.s (subsonic inflow/outflow)
       = 6   subsonic inflow - total pressure and temperature fixed
             subsonic outflow - back pressure specified
       = 8   tangency
       = 9   no-slip, adiabatic wall
       = 10  no-slip, T fixed at wall
       = 11  x-axis symmetry (j or k) - singular axis
       = 12  y-axis symmetry (j or k) - singular axis
       = 13  z-axis symmetry (j or k) - singular axis
       = 14  xy-plane axisymmetric    - singular axis
       = 15  yz-plane axisymmetric    - singular axis
       = 16  xz-plane axisymmetric    - singular axis
       = 17  reflection x-axis
       = 18  reflection y-axis
       = 19  reflection z-axis
       = 20  zonal boundary (fills q with adjacent zone values)
       = 21  axisymmetric wall (negative pi/80 angle)
       = 22  axisymmetric wall (positive pi/80 angle)
```

positive boundary type enforces full flux at boundary
negative boundary type splits the flux at boundary

NOTES: 1) Types 8,9,10 are enforced on the cell faces, the others are enforced at the cell centers.

2) Types 11, 12, 13 are typically used for grids that vary between 0 and 180 degrees in a cross-flow plane.

3) Types 14, 15, 16 are typically used for grids that are a pie-slice (axisymmetric) in the crossflow plane.

4) Types 17, 18, 19 should always be negative

5) Types 21, 22, should always be positive

6) Types 5 and 6 use perfect gas relations!

```
---------------------------------------------------------------------
--------------------------- DUMMY LINE ------------------------------
---------------------------------------------------------------------

******************** problem specification line 6 *********************
```

twall - wall temperature for boundary condition 10

pback - back pressure for boundary condition 6

```
     ttot   - total temperature for boundary condition 6

     ptot   - total pressure for boundary condition 6

     (UNITS ARE IMPORTANT)

-------------------------------------------------------------------------
---------------------------- DUMMY LINE ---------------------------------
-------------------------------------------------------------------------

******************** problem specification line 7 **********************

     Name of file containing boundary condition type for each point on
     all boundaries with type '0' above. and name of qfile for bc
     type 2's.  these filenames MUST be enclosed in single quotes

******************** end of problem specification **********************


########################## TIME INTEGRATION #############################


-------------------------------------------------------------------------
---------------------------- DUMMY LINE ---------------------------------
-------------------------------------------------------------------------

*********************** time integration line 1 ************************

     impl - time integration method flag
          = 0  explicit Runge-Kutta time integration
          = 1  LU decomposition in a on a line (only for 2-d problems)
          = 2  2-Factor AF in a plane
          = 3  3-Factor AF

     mstg   - number of stages for Runge-Kutta

     dt      - time step
          < 0 local time steps, cfl=abs(dt)
          > 0 constant time step = dt

     initdtl - flag for cfl calculation
          = 0 - use freestream q
          = 1 - use local values of q


-------------------------------------------------------------------------
---------------------------- DUMMY LINE ---------------------------------
-------------------------------------------------------------------------

*********************** time integration line 2 ************************

     irelu - reuse of LU decomposition flag
          = 0  - compute LU decomposition every time step
          = 1  - reuse LU decomposition based on tolreu & nremax
          = -NUM - begin reusing LU after NUM iterations

     nremax - maximum number of reuses before computing new LU decomp

     tolreu - tolerance of residual at which to begin reusing LU decomp

     initp  - only used if imarch=1
            = 0 fill new plane with free stream values
            = 1 fill new plane with converged solution of
              previous plane
```

```
sigma  - correction factor (~0.95) used in Vigneron technique
         for PNS

********************** end of time integration **********************

######################### INVISCID FLUXES ############################


-------------------------------------------------------------------
---------------------------- DUMMY LINE ----------------------------
-------------------------------------------------------------------

********************** inviscid fluxes line 1 ***********************

invflxi,invflxj,invflxk
                    inviscid flux flags for the i,j and k
                    directions respectively

                 = 0 no flux in this direction
                 = 1 steger-warming flux vector splitting
                 = 2 van leer flux vector splitting
                 = 3 roe's flux difference splitting
                 = 4 full flux with no splitting

invjaci, invjacj, invjack

                   this flag turns inviscid jacobians on and
                   off, note, the jacobian will be consistent
                   with the inflx chosen
                   (note steger-warming not yet implemented)
                   (note, interpretation different from 1.x)

********************** inviscid fluxes line 2 ***********************

limi,limj,limk    = 0 unlimited
                  = 1 smooth limiter
                  = 2 min-mod limiter
                  = 3 Spekraize-Venkat limiter (tuned to kappa=1/3)
                  = 4 ENO (second order using primitive variables)


rkapi,rkapj,rkapk - spatial discretization parameter
                  rkap <-1  or rkap > 1 first order upwind
                  = -1 fully upwind second order
                  = 1/3 upwind-biased third order

********************** end of inviscid fluxes ***********************

######################### VISCOUS FLUXES ############################


-------------------------------------------------------------------
---------------------------- DUMMY LINE ----------------------------
-------------------------------------------------------------------

*********************** viscous fluxes line 1 ***********************

visflxi, visflxj, visflxk - viscous flux flag for the i,j, and k
                            directions respectively
                            Can be +/- See NOTE
      = 0 inviscid
      = 1 laminar
      = 2 turbulent 0 wall
      = 3 turbulent dim wall
```

```
        = 4 turbulent both j walls

     NOTE: ivisci, iviscj, and ivisck can be chosen in various
           combinations of + or - values. '-' values give thin-layer
           contributions, '+' values in combination with other '+'
           values add to the thin-layer terms the cross-derivative
           contributions.

           Example, ivisci=-1,iviscj=1,ivisck=1 includes cross-
           in derivatives the j-k plane. Thus, ivisci,j,k=+1 gives
           Complete Navier-Stokes terms and would be a laminar
           calculation.  The choice ivisci=+1, iviscj=+4, ivisck=+4
           would also be a Complete Navier-Stokes calculation and
           would be a turbulent simulation in which the composite
           eddy viscosity is computed by combining distributions
           calculated from the j0,k0,jdim, and kdim surfaces.

     visjaci, visjacj, visjack - viscous flux jacobian for the i,j,
                                 and k directions respectively

        = 0 no jacobian in that direction
        = 1 thin layer jacobian in that direction

-------------------------------------------------------------------
--------------------------- DUMMY LINE ----------------------------
-------------------------------------------------------------------

*********************** viscous fluxes line 2 ***********************

     modlmu - laminar viscosity model
            = 1 Blottner curve fit for species laminar viscosity
            = 2 Sutherland formula for species laminar viscosity
            = 3 Keys Equation (perfect gas model only (0) valid for air!
            = 4 Helium tunnel equation w/Blottner curve fits
               (helium-n2-o2 model only (24))
            = 5 Helium tunnel equation w/Sutherland curve fits
               (helium-n2-o2 model only (24))

     modlk  - thermal conductivity model
            = 1 Euken relation for species thermal conductivity
            = 2 Sutherland formula for species thermal conductivity
            = 3 mixture thermal conductivity evaluated using a
              constant Prandtl number

     modld  - diffusion model
            = 1 simple binary diffusion

     prl    - Prandtl number (only used if modlk=3)

     prt    - turbulent Prandtl number

     scl    - laminar Schmidt number

     sct    - turbulent Schmidt number

-------------------------------------------------------------------
--------------------------- DUMMY LINE ----------------------------
-------------------------------------------------------------------

*********************** viscous fluxes line 3 ***********************

     ikeps  - flag for two equation model used
            = 0 - Baldwin-Lomax algebraic model
            = 1 - High Reynolds # model
            = 2 - Lam-Bremhorst model
            = 3 - Chien model
```

```
     ikejac - two equation Jacobian flag
            = 0 - no Jacobian
            = 1 - compute source term Jacobians

     NOTE: If you want to run a completely laminar calculation,
           ikeps must be 0.

     kemin   -  flag for imposing minimum value for K & e
               = 0 - do nothing
               = 1 - impose minimum value based on small percentage
                     of free stream values of K & epsilon

        NOTE: This adds stability, but the results may then be
              dependent on values of xkiv & tkelref chosen.  It
              is suggested that this be used in the early stages of
              iteration or with only a small value for xkiv.

     kefill  - flag to initialize TKE with eddy viscosity &
               mixing length from Baldwin-Lomax solution
            = 0 do nothing
            = 1 init with B_L solution

      ***  NOTE: Turn this off after initialization!!!!!  ***

*********************** end of viscous fluxes ***********************


###################### CHEMISTRY & THERMODYNAMICS ######################


-------------------------------------------------------------------
--------------------------- DUMMY LINE ----------------------------
-------------------------------------------------------------------

****************** chemistry & thermodynamics line 1 *****************

     itherm - thermodynamic model
            = 1 - equilibrium statistical mechanics model for species
                  internal energy ( translation + rotation + vibration )
            = 2 - curve fit for Cv (Gordon and McBride)
            = 3 - simplified vibrational relaxation model for species
                  internal energy
            = 4 - translation + rotation only (mixture of perfect gas)
            = 5 - liu equilibrium air curve fits
            = 6 - tgas equilibrium air curve fits

     chemfile - chemistry model file name in Inits/models.sum file
            =  'Perfect Gas' - Perfect Gas model
            =  'Kang & Dunn' - Kang and Dunn Air model
            =  'Park 1' - first Park Air model
            =  'Park 2' - second Park Air model
            =  'Park 3' - third Park Air model
            =  'Evans & Schexnayder 1' - first Evans & Schexnayder
                                         Hydrogen-Air model
            =  'Evans & Schexnayder 2' - second Evans & Schexnayder
                                         Hydrogen-Air model
            =  'Drummond 1' - first Drummond Hydrogen-Air model
            =  'Drummond 2' - second Drummond Hydrogen-Air model
            =  'NASP 1' - first NASP Hydrogen-Air model
            =  'NASP 2' - second NASP Hydrogen-Air model
            =  'NASP 3' - third NASP Hydrogen-Air model
            =  'NASP 4' - fourth NASP Hydrogen-Air model
            =  'Glass' - Glass model
            =  'Igra' - Igra model
            =  'Argon-Freon Plasma' - Argon-Freon Plasma model
```

```
        =  'Equilibrium Air' - Equilibrium Air model
        =  'Evans & Schexnayder 3' second Evans & Schexnayder
                                   hydrogen air model
                                   (Same reactions as 1.x)

    ieq -
        = 1 - frozen flow
        = 2 - chemical equilibrium flow
        = 3 - chemically reacting flow

    (Warning - ieq flag definitions  are different from 1.x)

    ichjac - chemical Jacobian flag
        = 0 - no Jacobian
        = 1 - compute chemical Jacobian (no temperature derivatives)
        = 2 - compute chemical Jacobian (full linearization)

****************** end of chemistry & thermodynamics *****************
```